Distributed and On-The-Fly Weight Computation for the Boltzmann Transport Equation

Heesoo Kim [1,3] Hector Carrillo Cabada[2,3]
Advisors: Jeffrey Haack[3](CCS-2), Jonas Lippuner[3](CCS-2)
[1]Brown University [2]University of New Mexico [3]Los Alamos National Laboratory

Boltzmann Transport Equation

The Boltzmann Transport Equation (BTE) describes the time evolution of the probability density function (PDF) of a particle being found at a given spatial location with a given momentum as the particles undergo elastic collisions. In this project, we focus on computing the collision operator Q in the BTE given by the following weighted convolution of Fourier transforms

$$\hat{Q}(\zeta) = \int_{\mathbb{R}^3} \hat{G}(\xi,\zeta)\hat{f}(\zeta-\xi)\hat{f}(\xi)d\xi \qquad (1)$$

where the convolutional weights $\hat{G}(\xi,\zeta)$ are given by

$$\hat{G}(\boldsymbol{\xi},\boldsymbol{\zeta}) = \left(\frac{\pi}{2}\right)^{3/2} \int_{\mathbb{R}^3} G(\mathbf{u},\boldsymbol{\zeta})e^{-i\xi\cdot\mathbf{u}}d\mathbf{u} \qquad (2)$$

Challenge

The computation time of and the memory required to store the weights scale as $\mathcal{O}\left(N^6\right)$, where N is the number of 1D velocity grid points:

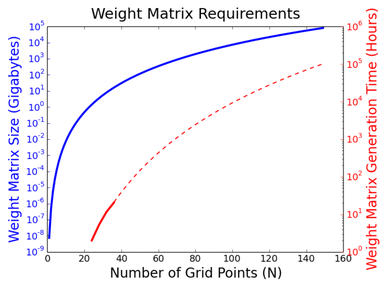

Figure 1: Weight matrix memory and time requirements vs number of points. Computation time doubles for every additional four N.

When modeling a mixture in which one species is more massive than the other, a large value of N is required to resolve both distributions.

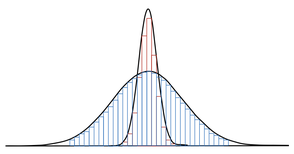

Figure 2: Particles with smaller mass have a wider PDF while particles with larger mass have a narrower PDF.

Due to the memory constraint for computations with large N we need to either distribute the precomputed weights across multiple nodes, or forgo storage of the weights and recompute them on the fly.

Methodology

We explore two methods for the convolutional weight computation and storage:

- **distribute the precomputed weights across several nodes using MPI**
- **calculate the weights on-the-fly on GPU(s) using CUDA.**

MPI+OpenMP and CUDA Workflow

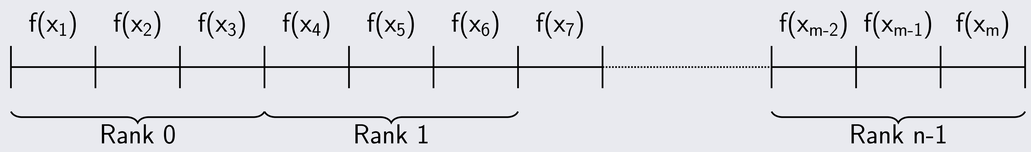

Figure 3: Original OpenMPI+MPI Implementation divides up the physical space, and each MPI rank takes in its own copy of the convolution weights and communicates adjacent cell information. This does not scale with the number of discretization.

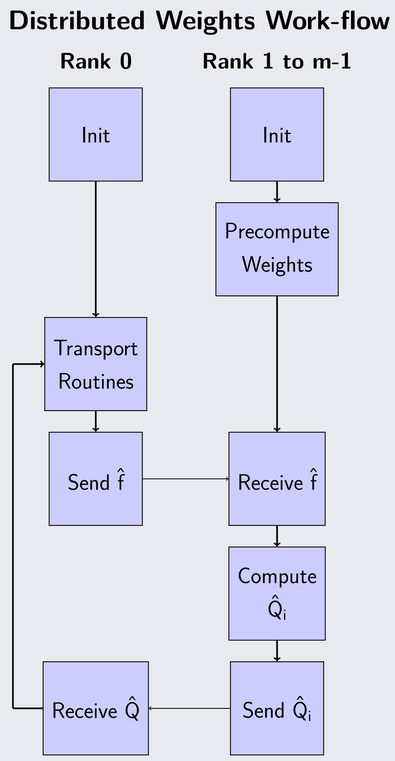

Figure 4: Each MPI rank precomputes a portion of the weights and computes only a portion of $\hat{Q}$ ($\hat{Q}_i$) at each time step. MPI communication is used collect the partial sums into $\hat{Q}$ and also to broadcast the updated $\hat{f}$.

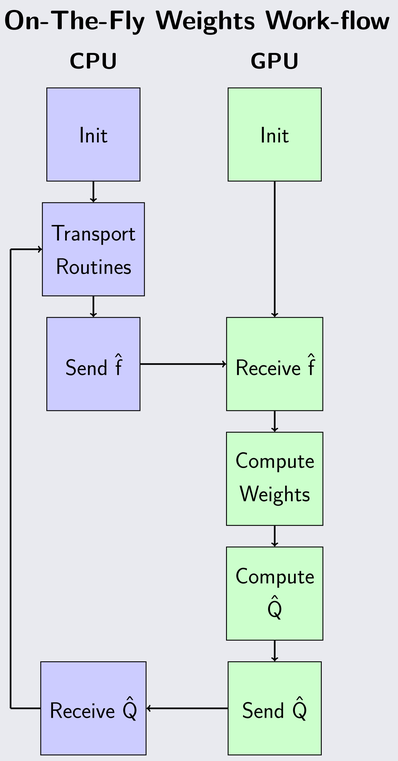

Figure 5: Instead of precomputing weights, the convolution weights and the associated sum is computed on the GPU to be passed back to CPU for updating the probability density function, $\hat{f}$.

Results + Discussion

While the new MPI work-flow performs comparably when executed on the same number of nodes and MPI ranks, its reduced memory requirements allows for larger runs.

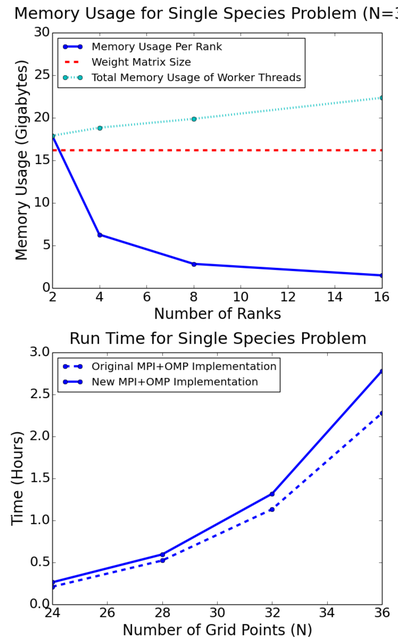

Figure 6: Left: memory usage vs. size of matrix; Right: computation time for 20 time steps of original and new MPI work-flow.

On the other hand, the CUDA work-flow has no memory requirements and is expected to perform faster as N grows when compared to the MPI runs using a small number of nodes.

References

[1] Jeffrey R. Haack.
A hybrid openmp and mpi implementation of a conservative spectral method for the boltzmann equation.
arXiv:1301.4195v1, 2013.

[2] Irene M. Gamba and Jeffrey R. Haack.
A conservative spectral method for the boltzmann equation with anisotropic scattering and the grazing collisions limit.
J. Comput. Phys., 2014.

Acknowledgements

- Support for this work was provided by U.S. Department of Energy at Los Alamos National Laboratory (LANL) supported by Contract No. DE-AC52-06NA25396.
- This project used resources of Darwin computing cluster at LANL and of Cori computing cluster at the National Energy Research Scientific Computing Center (NERSC).
- We are grateful to the LANL Parallel Computing Summer School and the mentors (Bob Robey, Hai Ah Nam, Kris Garrett) for their guidance and support.

LA-UR-18-26036